

本节内容

定点数 原码乘法运算

关注公众号【研途小时】获取后续课程完整更新！

雨声警告⚠️



今天的雨
下得跟依萍找她爸要钱那天一样大



今天的雨
下的跟祺贵人被打死那天一样大

关注公众号【研途小时】获取后续课程完整更新！

本节总览

乘法运算

乘法运算的实现思想

原码的一位乘法

补码的一位乘法

手算乘法（十进制）

r 进制: $K_n K_{n-1} \dots K_2 K_1 K_0 K_{-1} K_{-2} \dots K_{-m}$

$$= K_n \times r^n + K_{n-1} \times r^{n-1} + \dots + K_2 \times r^2 + K_1 \times r^1 + K_0 \times r^0 + K_{-1} \times r^{-1} + K_{-2} \times r^{-2} + \dots + K_{-m} \times r^{-m}$$

你怎么这个亚子



$$\begin{array}{r} 0.985 \\ \times 0.211 \\ \hline 985 \\ 985 \\ 1970 \\ \hline 0.207835 \end{array}$$



$$\begin{array}{r} 0.985 \\ \times 0.211 \\ \hline 0.000985 \\ 0.00985 \\ 0.1970 \\ \hline 0.207835 \end{array}$$



$$0.211 = 2 \times 10^{-1} + 1 \times 10^{-2} + 1 \times 10^{-3}$$

$$0.985 = 985 \times 10^{-3}$$

$$0.985 \times 0.211 = (985 \times 1 \times 10^{-6}) + (985 \times 1 \times 10^{-5}) + (985 \times 2 \times 10^{-4})$$

手算乘法（二进制）

r 进制: $K_n K_{n-1} \dots K_2 K_1 K_0 K_{-1} K_{-2} \dots K_{-m}$

$$= K_n \times r^n + K_{n-1} \times r^{n-1} + \dots + K_2 \times r^2 + K_1 \times r^1 + K_0 \times r^0 + K_{-1} \times r^{-1} + K_{-2} \times r^{-2} + \dots + K_{-m} \times r^{-m}$$

$$\begin{array}{r} 0.1101 \\ \times 0.1011 \\ \hline 1101 \\ 1101 \\ 0000 \\ 1101 \\ \hline 0.10001111 \end{array}$$



$$\begin{array}{r} 0.1101 \\ \times 0.1011 \\ \hline 0.00001101 \\ 0.0001101 \\ 0.000000 \\ 0.01101 \\ \hline 0.10001111 \end{array}$$



考虑用机器实现:

- 实际数字有正负, 符号位如何处理?
- 乘积的位数扩大一倍, 如何处理?
- 4个位积都要保存下来最后统一相加?

(乘数) $0.1011 = 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4}$

(被乘数) $0.1101 = 1101 \times 2^{-4}$

用“移位”实现

$$0.1101 \times 0.1011 = (1101 \times 1 \times 2^{-8}) + (1101 \times 1 \times 2^{-7}) + (1101 \times 0 \times 2^{-6}) + (1101 \times 1 \times 2^{-5})$$

关注公众号【研途小时】获取后续课程完整更新 !

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

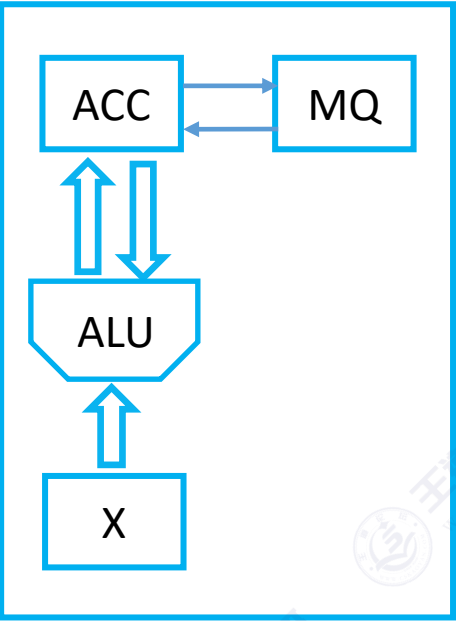
数值位取绝对值进行乘法计算

$[|x|]_{\text{原}} = 0.1101$ ， $[|y|]_{\text{原}} = 0.1011$

穿越：运算器的基本组成



运算器



运算器：用于实现算术运算（如：加减乘除）、逻辑运算（如：与或非）

- ACC: 累加器，用于存放操作数，或运算结果。
- MQ: 乘商寄存器，在乘、除运算时，用于存放操作数或运算结果。
- X: 通用的操作数寄存器，用于存放操作数
- ALU: 算术逻辑单元，通过内部复杂的电路实现算数运算、逻辑运算

Accumulator
Multiple-Quotient Register
Arithmetic and Logic Unit

	加	减	乘	除
ACC	被加数、和	被减数、差	乘积高位	被除数、余数
MQ			乘数、乘积低位	商
X	加数	减数	被乘数	除数

关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行乘法计算

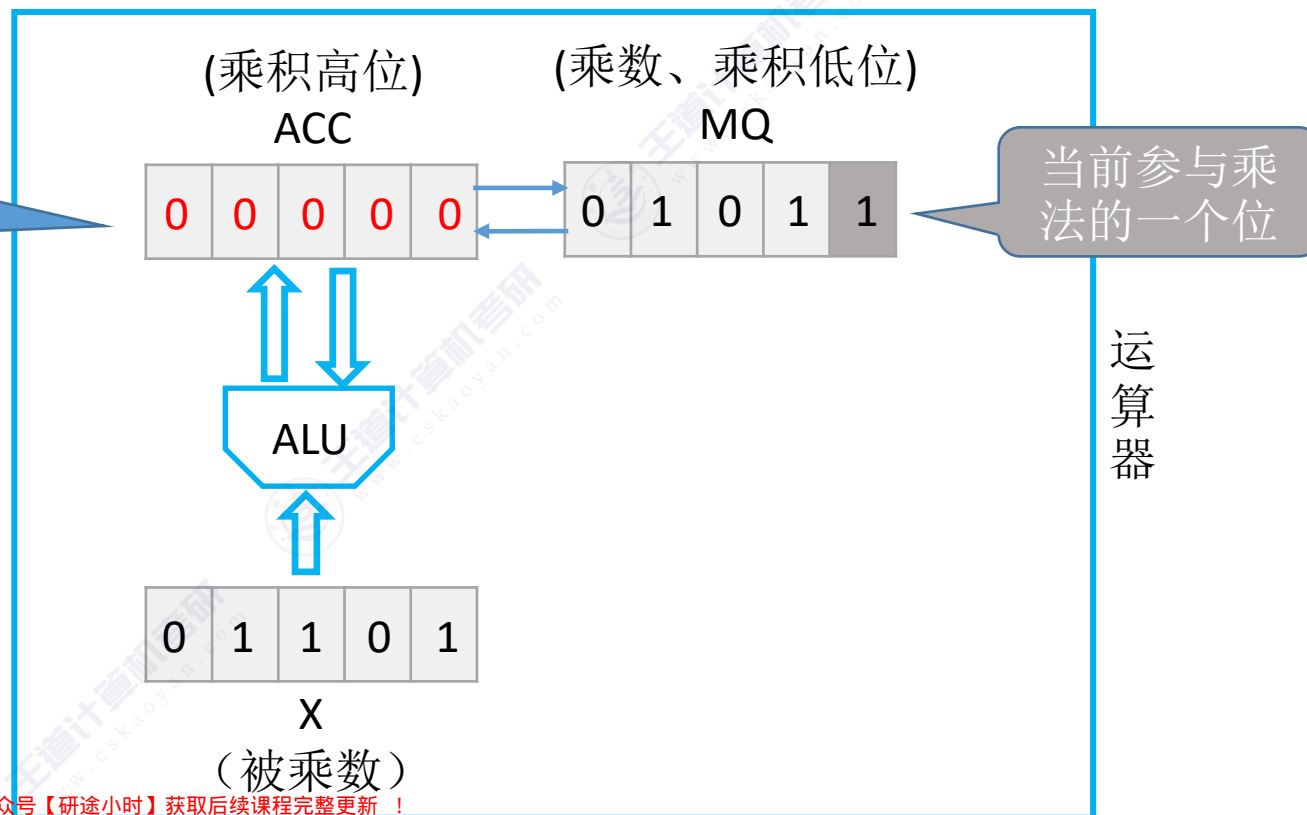
实现方法：先加法再移位，重复 n 次

```
  0.1101
× 0.1011
-----
  01101
  01101
  00000
  01101
-----
0.10001111
```

在正式进行乘法之前，ACC置0

$00000 + 01101 = 01101$

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行乘法计算

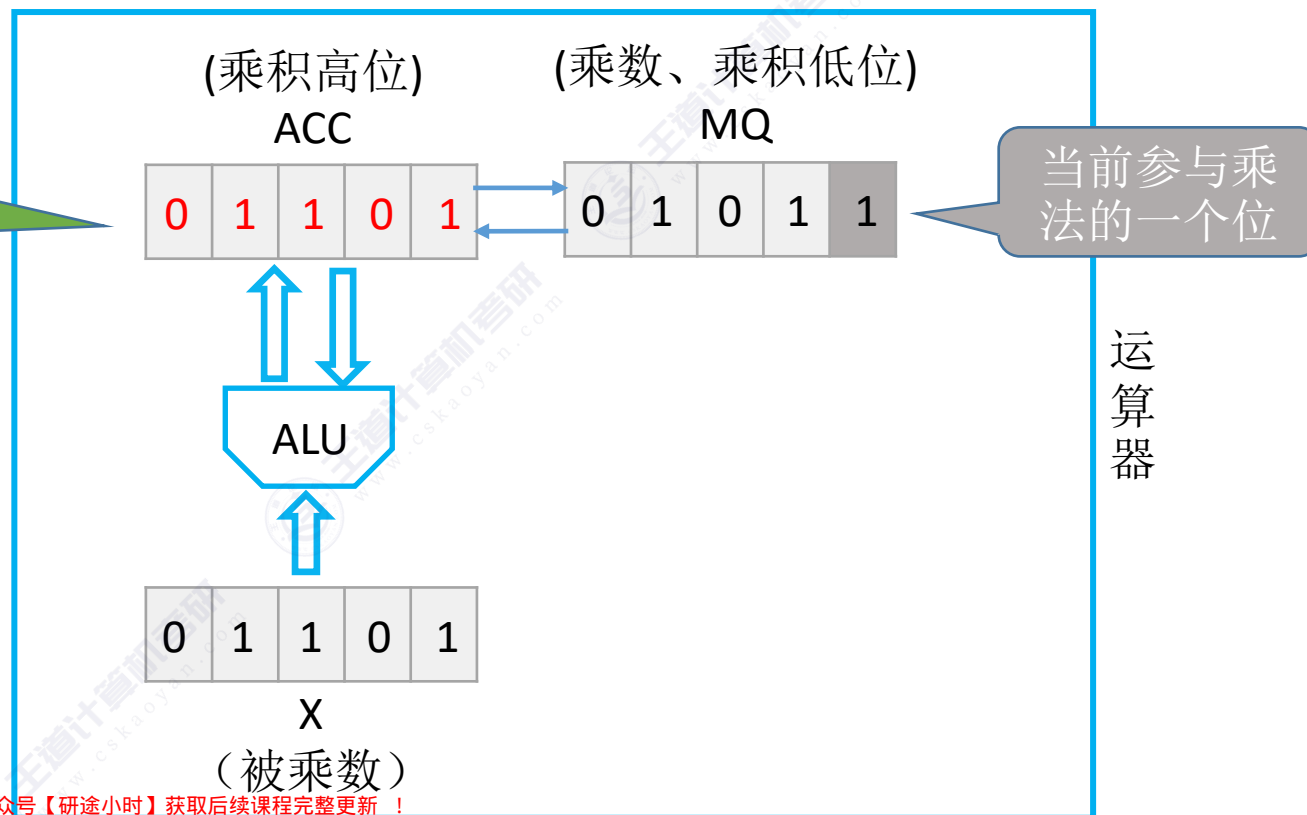
实现方法：先加法再移位，重复 n 次

```
  0.1101
× 0.1011
-----
  01101
  01101
  00000
  01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0

(ACC)+(X)→ACC

00000+01101=01101



原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

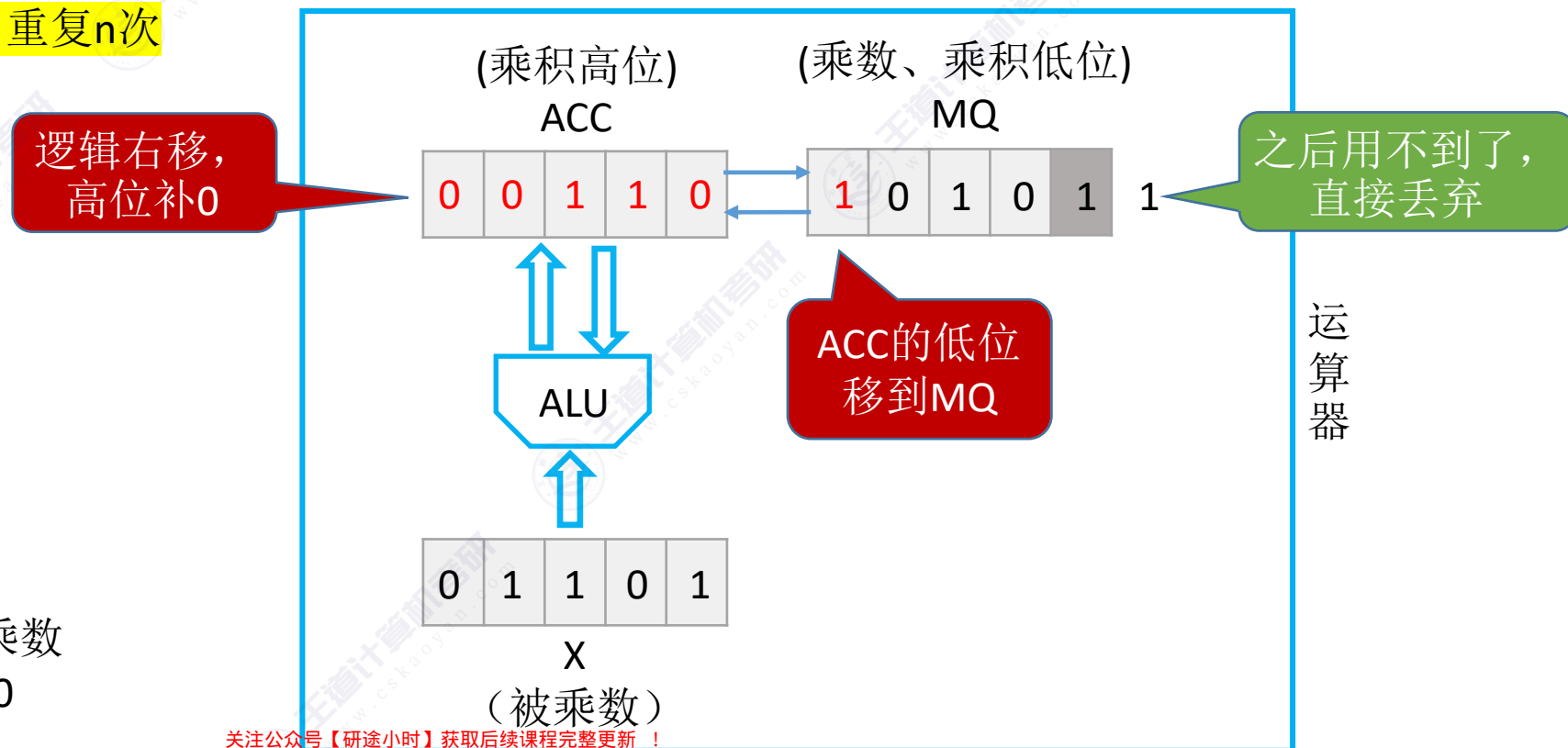
符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行乘法计算

实现方法：先加法再移位，重复n次

```
  0.1101
× 0.1011
-----
  01101
 01101
00000
01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行乘法计算

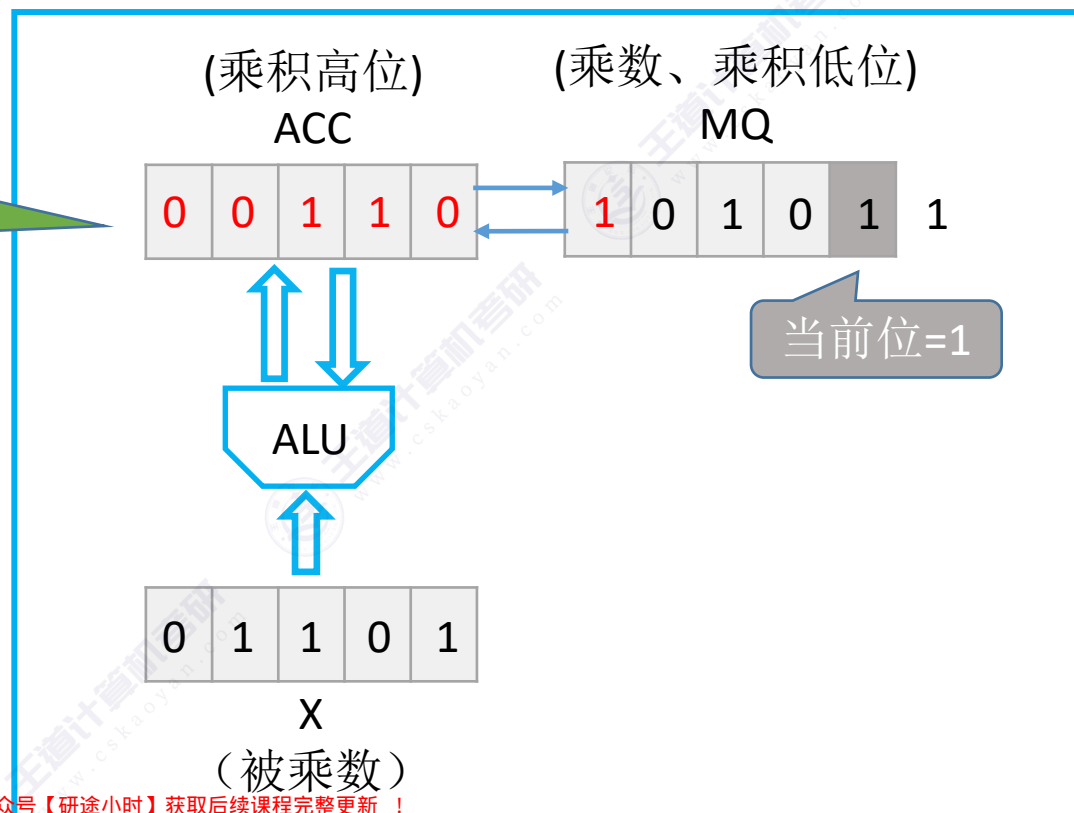
实现方法：先加法再移位，重复n次

```
  0.1101
× 0.1011
-----
  01101
  01101
  00000
  01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0

(ACC)+(X)→ACC

00110+01101=10011



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行乘法计算

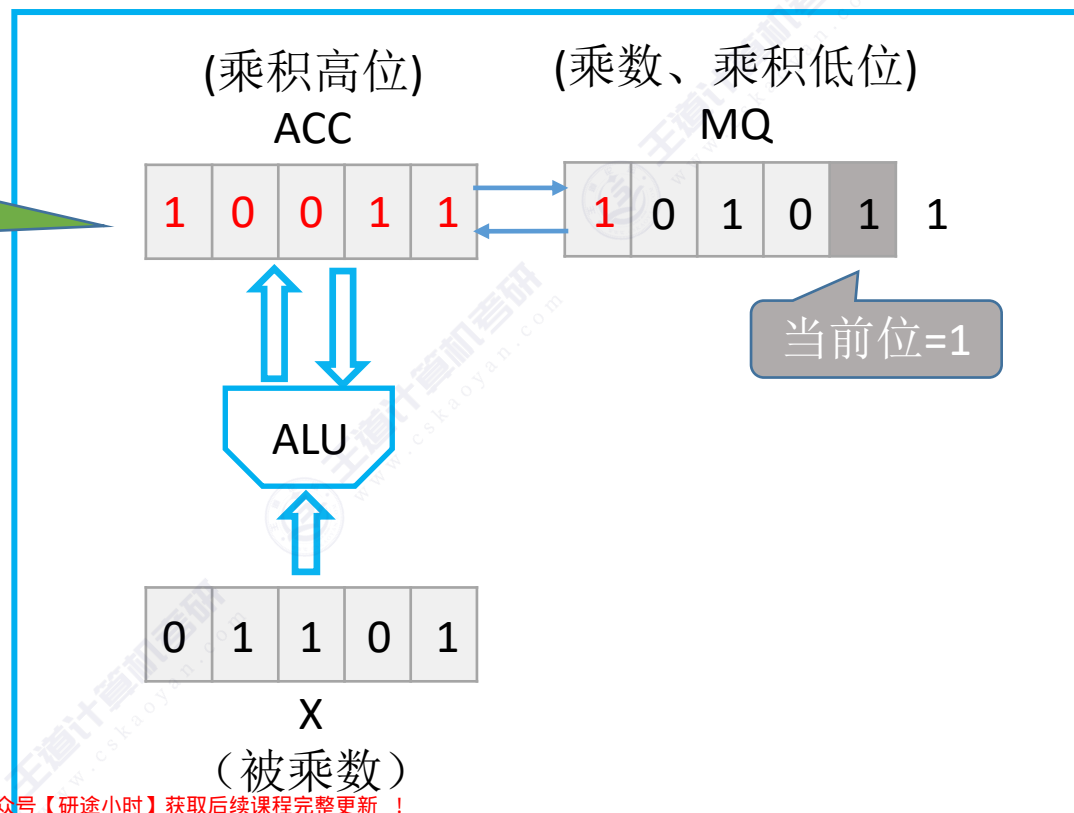
实现方法：先加法再移位，重复 n 次

```
  0.1101
× 0.1011
-----
  01101
  01101
  00000
  01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0

(ACC)+(X)→ACC

00110+01101=10011



运算器

关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

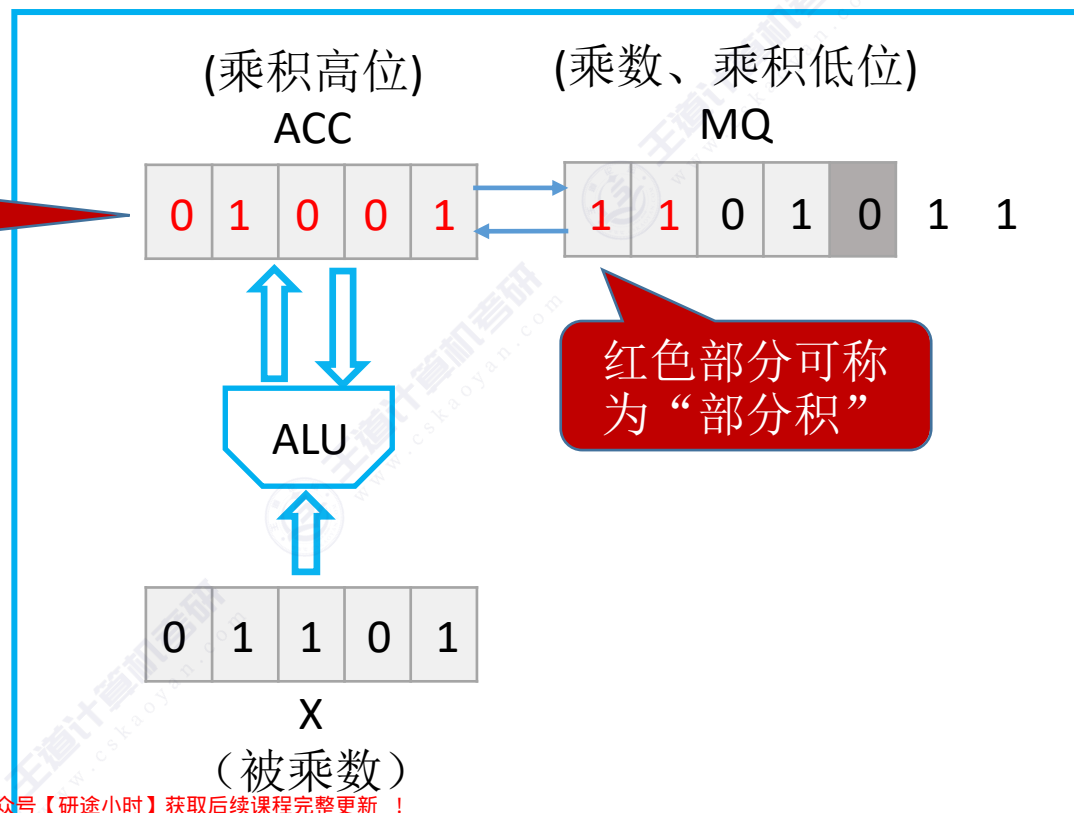
数值位取绝对值进行乘法计算

实现方法：先加法再移位，重复 n 次

```
  0.1101
× 0.1011
-----
  01101
 01101
00000
01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0

逻辑右移，
高位补0



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

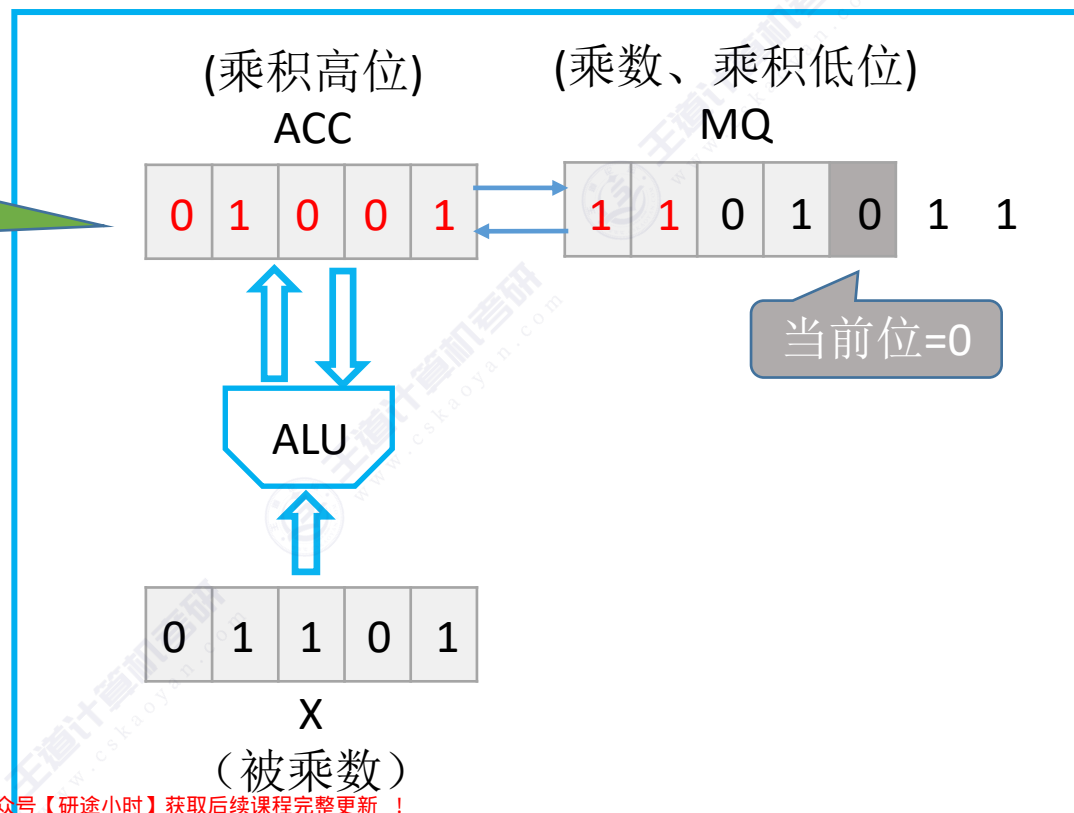
数值位取绝对值进行乘法计算

实现方法：先加法再移位，重复 n 次

```
  0.1101
× 0.1011
-----
  01101
 01101
 00000
 01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0

(ACC)+0→ACC



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

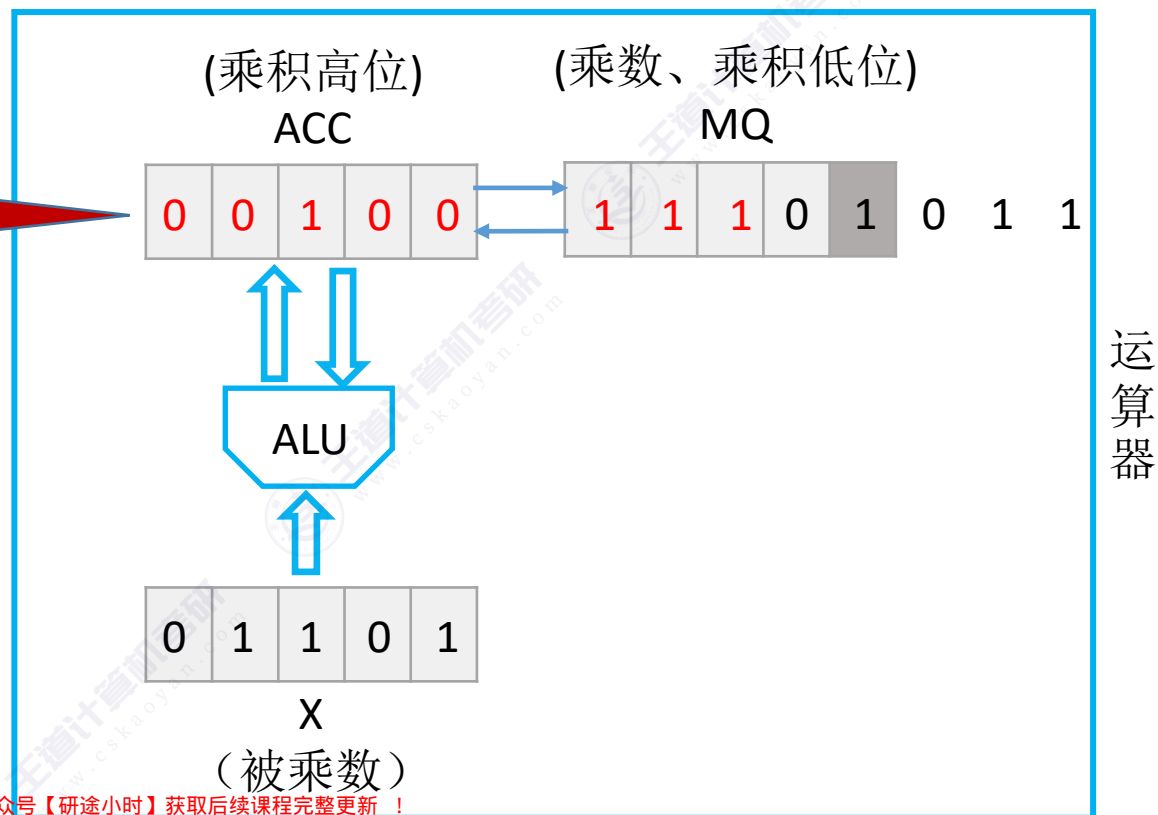
数值位取绝对值进行乘法计算

实现方法：先加法再移位，重复 n 次

```
  0.1101
× 0.1011
-----
  01101
  01101
  00000
  01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0

逻辑右移，
高位补0



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行乘法计算

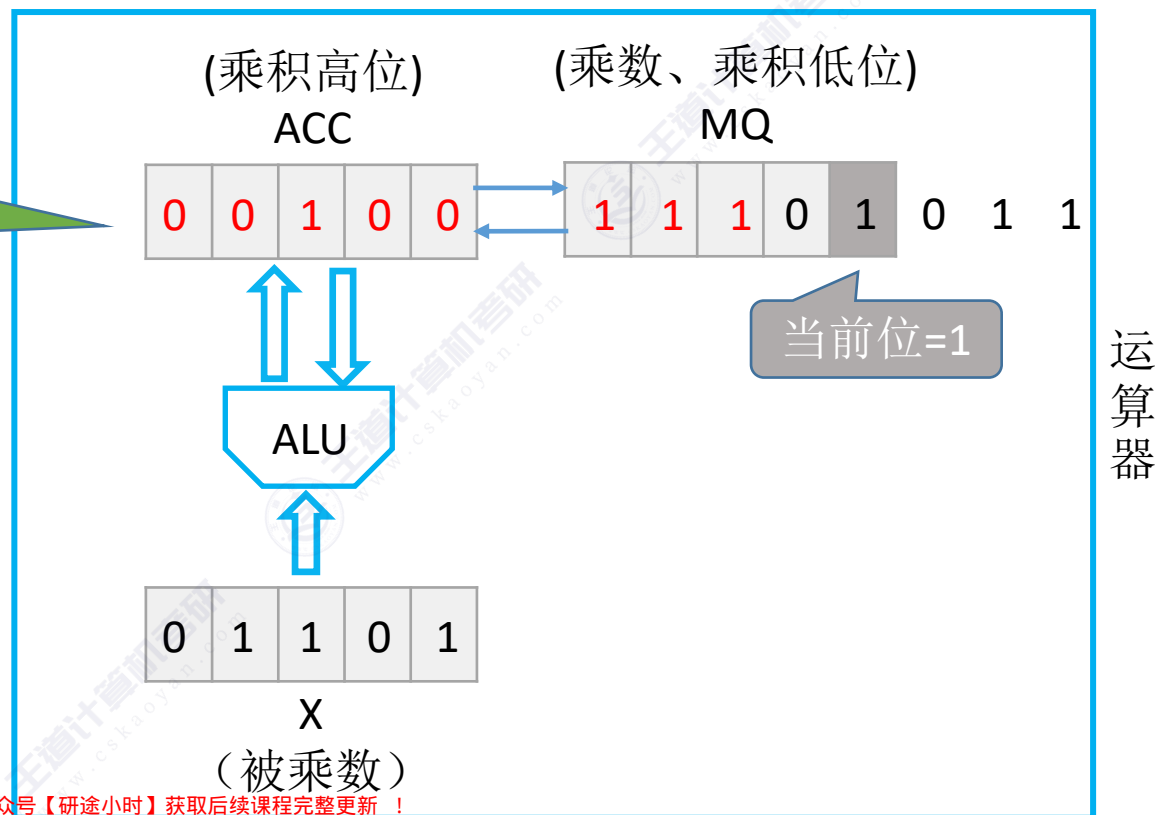
实现方法：先加法再移位，重复 n 次

```
  0.1101
× 0.1011
-----
  01101
  01101
  00000
  01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0

(ACC)+(X)→ACC

00100+01101=10001



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行乘法计算

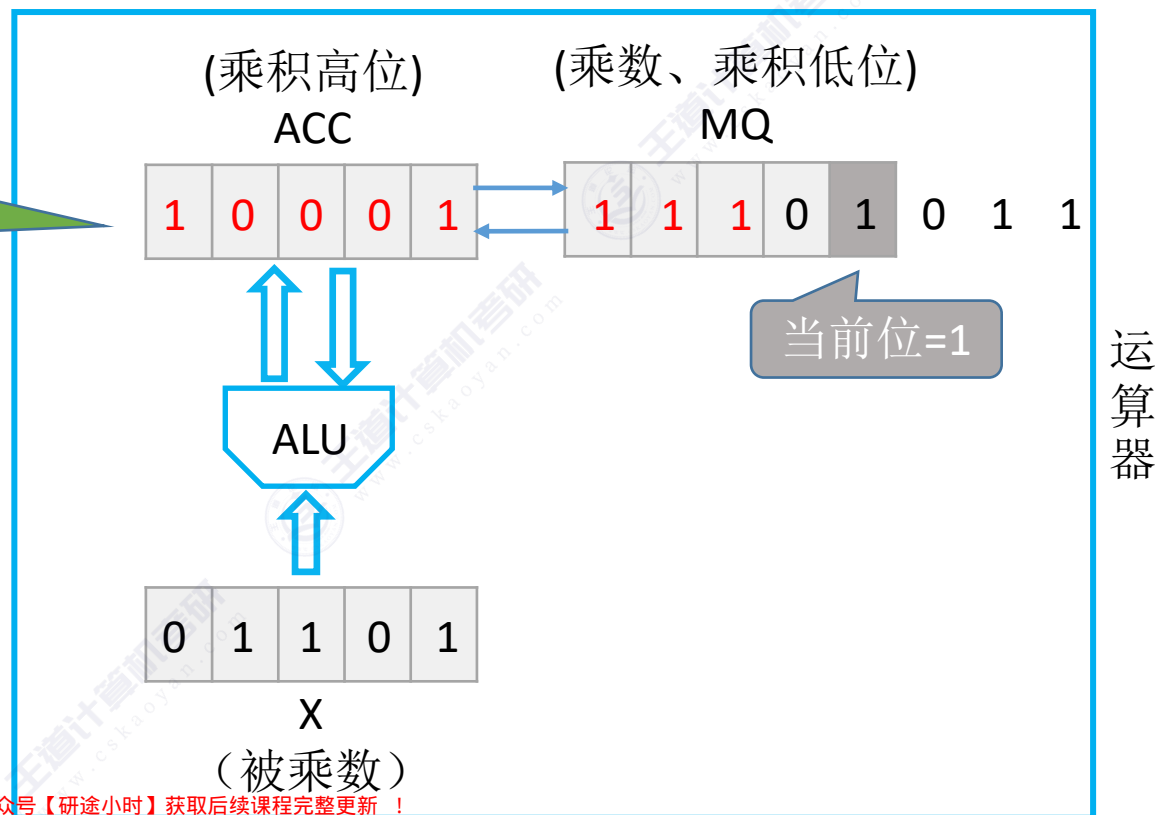
实现方法：先加法再移位，重复 n 次

```
  0.1101
× 0.1011
-----
  01101
  01101
  00000
  01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0

(ACC)+(X)→ACC

00100+01101=10001



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

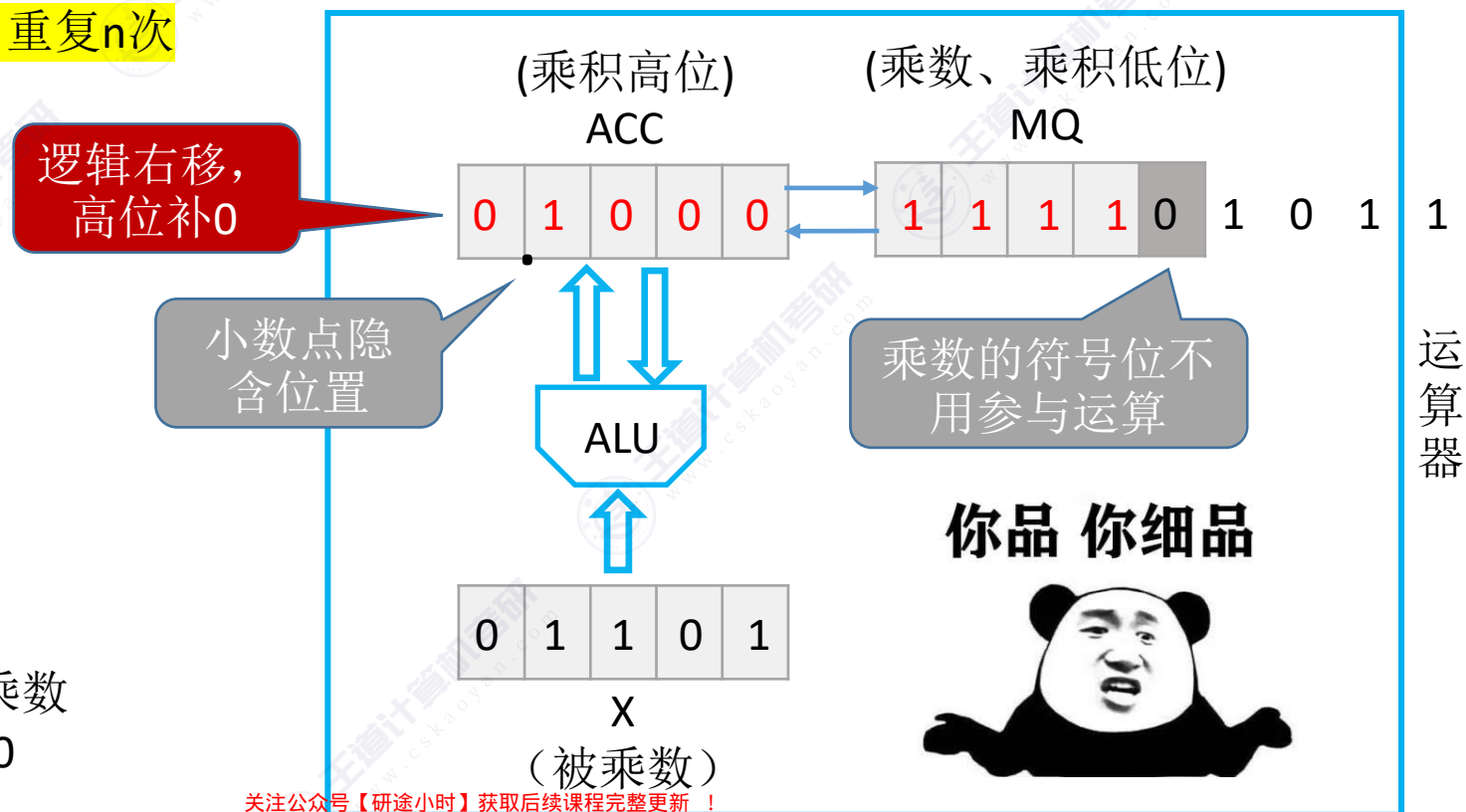
符号单独处理：符号位 = $x_s \oplus y_s$

数值位取绝对值进行乘法计算

实现方法：先加法再移位，重复 n 次

```
  0.1101
×  0.1011
-----
  01101
  01101
  00000
  01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法

设机器字长为 $n+1=5$ 位（含1位符号位）， $[x]_{\text{原}} = 1.1101$ ， $[y]_{\text{原}} = 0.1011$ ，采用原码一位乘法求 $x \cdot y$

符号位	数值位
-----	-----

符号单独处理：符号位 = $x_s \oplus y_s$

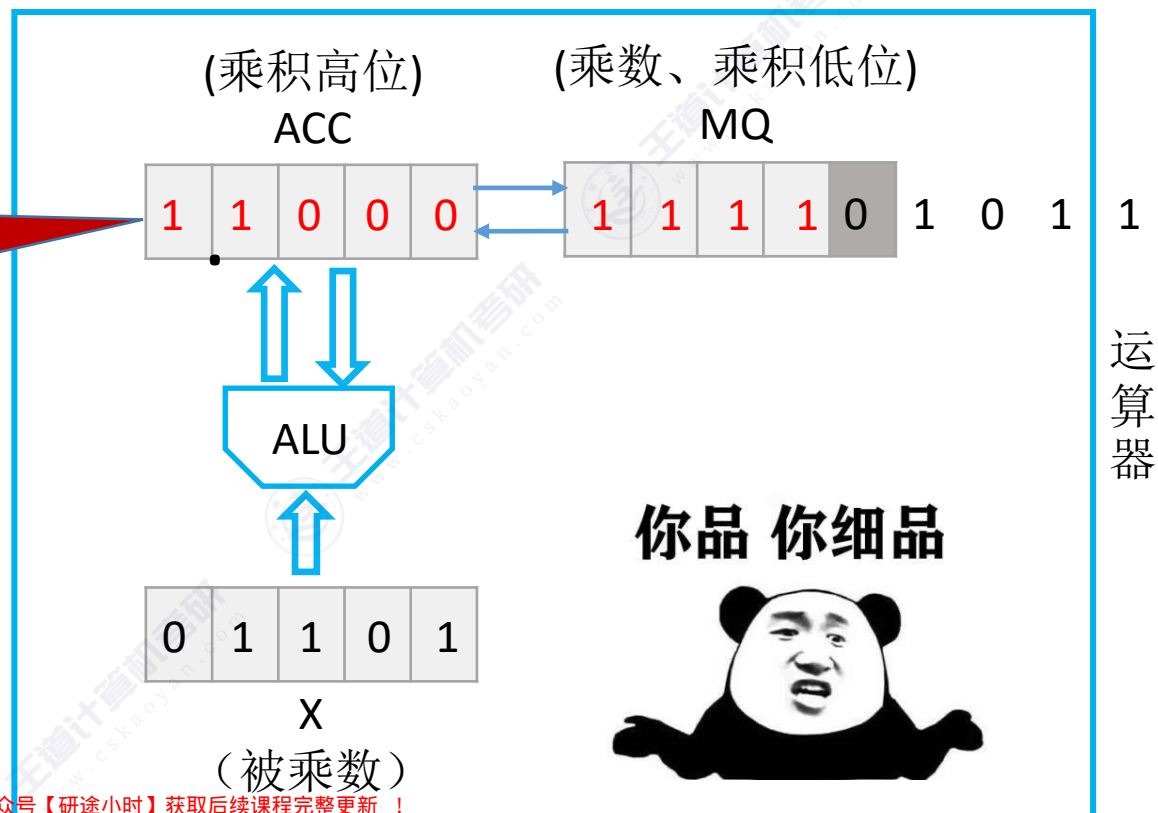
数值位取绝对值进行乘法计算

实现方法：先加法再移位，重复n次

```
  0.1101
× 0.1011
-----
  01101
  01101
  00000
  01101
-----
0.10001111
```

当前位=1，则ACC加上被乘数
当前位=0，则ACC加上 0

修改符号
位 $x_s \oplus y_s = 1$



关注公众号【研途小时】获取后续课程完整更新！

原码一位乘法（手算模拟）

设机器字长为5位（含1位符号位， $n=4$ ）， $x = -0.1101$ ， $y = +0.1011$ ，采用原码一位乘法求 $x \cdot y$



解： $|x| = 00.1101$ ， $|y| = 00.1011$ ，原码一位乘法的求解过程如下。

MQ

ACC

(高位部分积)

(低位部分积/乘数)

说明

通用寄存器

$+|x|$

00.0000

1011

丢失位

起始情况

$C_4=1$ ，则 $+|x|$

00.1101

00.1101

右移

00.0110

----- 1101

1

右移部分积和乘数

$C_4=1$ ，则 $+|x|$

$+|x|$

00.1101

01.0011

右移

00.1001

----- 1110

11

右移部分积和乘数

$C_4=0$ ，则 $+0$

$+0$

00.0000

00.1001

右移

00.0100

----- 1111

011

右移部分积和乘数

$C_4=1$ ，则 $+|x|$

$+|x|$

00.1101

01.0001

右移

00.1000

----- 1111

1011

右移部分积和乘数

乘数全部移出

结果的绝对值部分

Tips:

- 乘数的符号位不参与运算，可以省略
- 原码一位乘可以只用单符号位
- 答题时最终结果最好写为原码机器数

原码一位乘法：机器字长 $n+1$ ，数值部分占 n 位

符号位通过异或确定；数值部分通过被乘数和乘数绝对值的 n 轮加法、移位完成
根据当前乘数中参与运算的位确定(ACC)加什么。若当前运算位=1，则 $(ACC)+[|x|]$ 原；
若=0，则 $(ACC)+0$ 。

每轮加法后ACC、MQ的内容统一逻辑右移

符号位 $P_s = x_s \oplus y_s = 1 \oplus 0 = 1$ ，得 $x \cdot y = -0.10001111$ 。

关注公众号【研途小时】获取后续课程完整更新！